

Memahami Penggunaan Diagram Arus Data

Haviluddin

Program Studi Ilmu Komputer, FMIPA Universitas Mulawarman

Jl. Barong Tongkok no.5 Kampus Unmul Gn. Kelua Sempaja Samarinda 75119

Abstrak

Diagram Arus Data atau yang sering disebut sebagai *Data Flow Diagram* (DFD) merupakan alat perancangan sistem yang berorientasi pada alur data dengan konsep dekomposisi dapat digunakan untuk penggambaran analisa maupun rancangan sistem yang mudah dikomunikasikan oleh profesional sistem kepada pemakai maupun pembuat program.

Diagram Arus Data mempunyai empat komponen utama yaitu entitas (*entities*), proses (*process*), media penyimpanan (*data storage*) dan arus data (*data flows*). Ke empat komponen tersebut menggambarkan arus sistem agar mudah dipahami.

Kata Kunci : Data flow diagram, entitas, proses, media penyimpanan, arus data

PENDAHULUAN

Sebagian besar para perancang sistem informasi dalam menggambarkan informasi dengan memanfaatkan Diagram Arus Data (DAD), pemilihan metode perancangan sistem ini berdasar kepada prinsip

1. DAD tersebut mudah dipahami oleh analisis maupun orang awam.
2. DAD menggambarkan level sistem tingkat tinggi, beserta batasan-batasan sistem dan informasi arus datanya.
3. DAD menggambarkan secara rinci sistem yang ada di setiap komponennya.

Diagram Arus Data (DAD) sendiri digunakan untuk memberikan gambaran sistem secara keseluruhan hingga batasan sistem, sumber-sumber dan tujuan data, proses data, arus data dan media penyimpanan dengan memanfaatkan simbol-simbol dalam DAD.

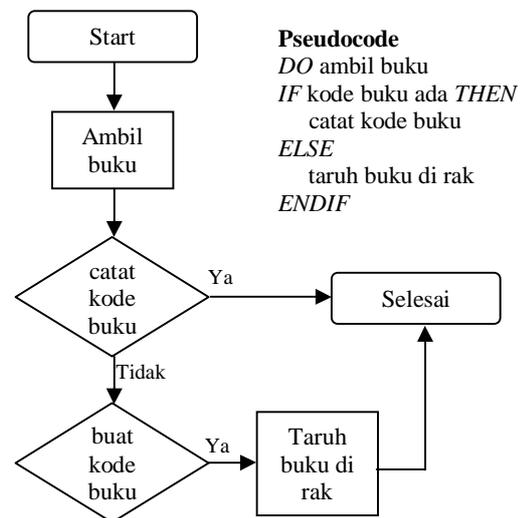
Ini bersesuaian dengan pengertian DAD adalah penggambaran sistem yang direpresentasikan secara grafik (Gelinis *et.al.*, 2004).

Sehingga DAD ini dapat digunakan untuk penggambaran analisa maupun rancangan sistem yang mudah dikomunikasikan oleh profesional sistem kepada pemakai maupun pembuat program.

Flowchart dan Pseudocode

Dalam merancang pembangunan suatu program aplikasi, para programmer menggunakan kombinasi *flowchart* dan *pseudocode*. *Pseudocode* biasanya lebih mudah dipahami daripada *flowchart*.

Gambar 1 memperlihatkan *flowchart* dan *pseudocode* untuk mencatat kode buku dalam sistem informasi perpustakaan.



Gambar 1. Flowchart dan Pseudocode dalam Sistem Informasi Perpustakaan

Berbeda dengan DAD, Hirarki di dalamnya menggambarkan hubungan sistem dengan lingkungan luarnya. Dimulai dengan level 0 atau disebut Context Diagram ialah diagram level tertinggi kemudian diikuti oleh level-level berikutnya yang juga merepresentasikan bagian dari sistem.

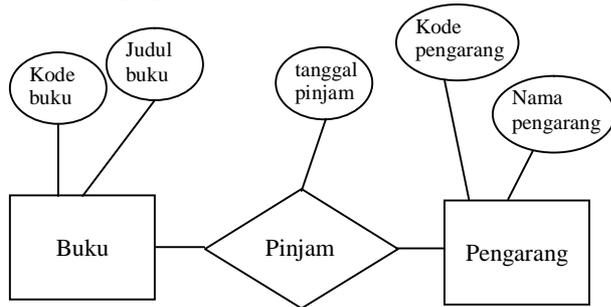
Entity-Relationship Diagrams (ERDs)

Entity-Relationship Diagrams (ERDs) merupakan suatu model untuk menjelaskan hubungan antar data dalam basis data berdasarkan persepsi bahwa *real world* terdiri dari objek-objek dasar yang mempunyai hubungan antar objek-objek dasar tersebut yang dilukiskan dengan menggunakan komponen-komponen data entitas, atribut dan relasi antar entitas (Ryan K.S and Ronald R.P. 2001)

Bila dilihat pada pengertian di atas, Entity-Relationship Diagrams (ERDs) adalah model objek data untuk menggambarkan arus informasi pada peringkat logika secara abstrak.

ERD model yang dibuat berdasarkan anggapan bahwa dunia nyata terdiri dari koleksi obyek-obyek dasar yang dinamakan entitas (*entity*) serta hubungan (*relationship*) antara entitas-entitas itu.

Gambar 2 memperlihatkan ERD dalam sistem informasi perpustakaan.



Gambar 2. ERD dalam Sistem Informasi Perpustakaan

DIAGRAM ARUS DATA (DAD)

DAD pada dasarnya sebuah diagram yang menjelaskan bagaimana hubungan bersama dari bagian file, laporan, sumber dokumen dan sebagainya, DAD termasuk alat komunikasi medium yang baik antara designer dan pemakai karena mudah dipahami (Kendall & Kendall, 2002).

Gambaran grafis dari suatu sistem yang menggunakan sejumlah bentuk-bentuk simbol untuk menggambarkan bagaimana data mengalir melalui suatu proses yang saling berkaitan. (McLeod, 2001).

Dapat diartikan Diagram Arus Data (DAD) menggantikan fungsi *flowcharts* dan *pseudocode* sebagai alat bantu pilihan untuk mendisain program yang berorientasi pada alur data dengan konsep dekomposisi.

Beberapa istilah yang berkenaan dengan DAD seperti Data Flow Diagram (DFD), Bubble Chart, Bubble Diagram, Model Proses, Diagram Alur Kerja, atau Model Fungsi.

KOMPONEN-KOMPONEN DAD

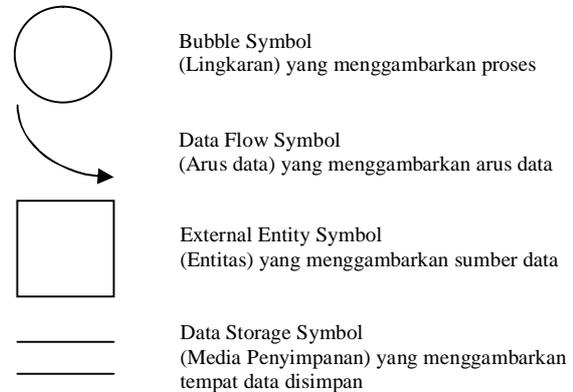
Sejauh ini banyak pakar menggunakan lebih banyak komponen dalam perancangan suatu sistem untuk menjelaskan sistemnya.

DAD hanya memiliki empat komponen dasar dalam mengilustrasikan bagaimana aliran data dalam suatu sistem, komponen tersebut adalah entitas (*entities*), proses (*process*), media penyimpanan (*data storage*) dan arus data (*data flows*).

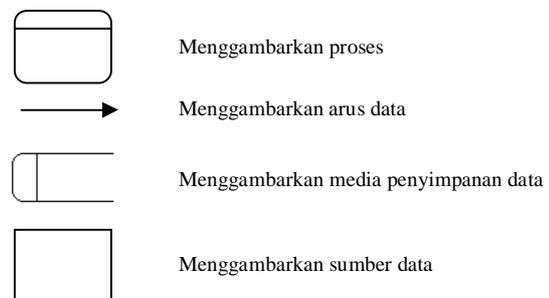
Sebelum dijelaskan lebih jauh mengenai komponen-komponen tersebut, sejauh ini tidak ada kesepakatan ahli tentang suatu simbol yang

digunakan, tetapi setiap simbol yang digunakan mempunyai karakteristik yang berbeda.

Gelinas *et.al.*, 2004, menggambarkan komponen-komponen DAD dengan;



Ryan K.S and Ronald R.P. 2001, menggambarkan komponen-komponen DAD dengan;



Berikut akan dijelaskan komponen-komponen tersebut.

a. Entitas (Entity)

Entitas merupakan sumber data atau menjadi tujuan data. Sumber atau tujuan tersebut dibedakan dengan garis panah atau keterangan yang berkaitan. Istilah lain dari entitas adalah terminator. Terdapat dua jenis terminator; yaitu terminator sumber (*source*) yang merupakan terminator yang menjadi sumber; dan terminator tujuan (*sink*) yang merupakan terminator yang menjadi tujuan data / informasi sistem.

b. Proses (Process)

Proses adalah suatu manipulasi terhadap data, menggambarkan bagian dari sistem yang mentransformasikan input menjadi output. Pemberian nama proses dilakukan dengan menggunakan kata kerja transitif (kata kerja yang membutuhkan obyek) dan setiap prosesnya diberikan nama dan nomor proses, seperti 1.1. Menghitung Peminjam, 2.1. Mencetak Daftar Buku.

Ada empat kemungkinan yang dapat terjadi dalam proses sehubungan dengan input dan output yaitu; satu input menghasilkan satu output, satu input

menghasilkan banyak output, banyak input menghasilkan satu output; dan banyak input menghasilkan banyak output.

c. Media Penyimpanan (*Data Store*)

Komponen ini digunakan untuk membuat model sekumpulan paket data dan diberi nama dengan kata benda jamak, misalnya *Buku*. Data store ini biasanya berkaitan dengan penyimpanan-penyimpanan, seperti file atau database yang berkaitan dengan penyimpanan secara komputerisasi, misalnya file disket, file harddisk, file pita magnetik. Data store juga berkaitan dengan penyimpanan secara manual seperti buku alamat, file folder, dan agenda.

Suatu data store dihubungkan dengan alur data hanya pada komponen proses, tidak dengan komponen DAD lainnya. Alur data dibedakan menjadi dua macam yaitu alur data dari data store yang berarti sebagai pembacaan atau pengaksesan data; dan alur data ke data store yang berarti sebagai pengupdatean data, seperti menambah data baru, menghapus, atau mengubah/ memodifikasi data. Dengan kata lain, proses alur data bertanggung jawab terhadap perubahan yang terjadi pada data store.

d. Arus Data (*Data Flow*)

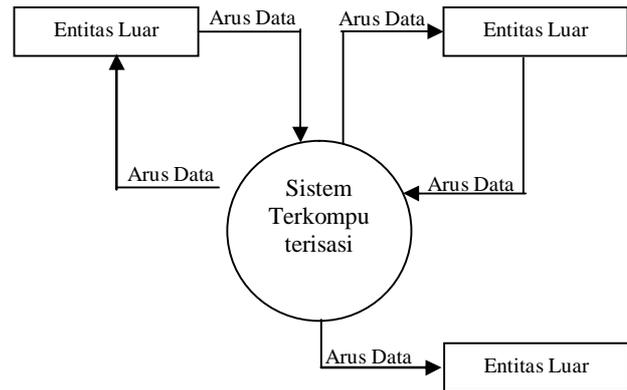
Sekelompok elemen data yang berhubungan secara logis yang bergerak dari satu titik atau proses ke titik atau proses yang lain. (McLeod, 2001).

Suatu arus data digambarkan dengan anak panah, yang menunjukkan arah menuju ke dan keluar dari suatu proses. Arus data ini digunakan untuk menerangkan perpindahan data atau paket data/informasi dari satu bagian sistem ke bagian lainnya.

PROSES MEMBANGUN DAD

Untuk menghasilkan suatu gambaran sistem yang baik dan terdokumentasi maka DAD dibangun dengan proses bertingkat. Tingkatan DAD berdasarkan **level-*n* diagram** yang dimulai dari level 0.

Secara umum proses membangun DAD tidak terlepas dari proses pengambilan data, tetapi penekanannya ada pada proses, yang bertujuan untuk menentukan entitas, proses, media penyimpanan dan arus data. Gambar 3 memperlihatkan Diagram Kontek yang menyatakan proses dan media penyimpanan secara umum.



Gambar 3. Diagram Kontek

Adapun penggambaran level-level berikutnya dalam DAD bertujuan untuk memperjelas proses data.

Rambu-rambu Membuat DAD

Adanya rambu-rambu dalam DAD digunakan untuk memudahkan para perancang sistem dalam mengimplementasikan hasil pengambilan data sehingga efektivitas DAD akan terlihat, menghindari pembentukan DAD yang salah atau DAD yang tidak lengkap atau tidak konsisten secara logika.

McLeod, 2001. Menetapkan untuk pemberian label tiap arus data dengan nama yang unik; konsisten dengan nama arus data; tunjukkan penempatan yang tepat bagi catatan-catatan yang dihapuskan dari penyimpanan data; tidak menyertakan proses pemrograman saat dokumentasi; hindari proses baca saja dan proses baca diijinkan jika waktu berfungsi sebagai pemicu.

Sedangkan menurut Jeffrey *et.al.*, 2002. Menetapkan proses input dan output harus dibedakan dengan jelas; arus data harus diberi nama unik.

Kesalahan umum dalam melakukan proses di dalam DAD adalah; adanya proses mempunyai input tetapi tidak menghasilkan output. Kesalahan ini disebut dengan **black hole** (lubang hitam), karena data masuk ke dalam proses dan tidak terlihat keluarannya. Kemudian proses menghasilkan output tetapi tidak pernah menerima input. Kesalahan ini disebut dengan **miracle** (ajaib), karena output yang dihasilkan tanpa pernah menerima input.

a. Menentukan Batasan Sistem

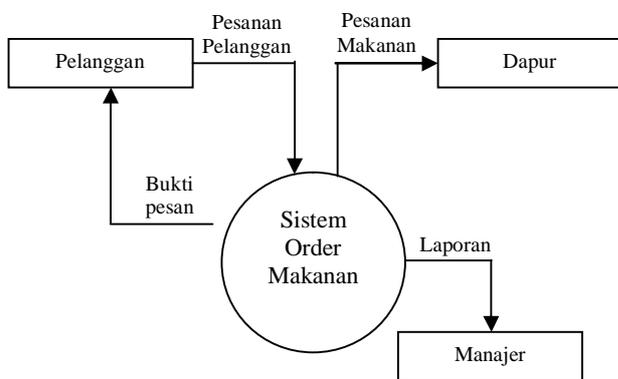
Langkah pertama yang dilakukan adalah menentukan item-item data apa saja yang akan ada dalam sistem, walaupun hal ini tidak selalu dikerjakan oleh perancang sistem. Untuk hal ini biasanya diatasi dengan membentuk tabel dimana dapat memperlihatkan sumber dan tujuan data.

Batasan sistem harus ditentukan terlebih dahulu bertujuan agar pemakai mengetahui dengan lingkungan mana saja sistem mereka berhubungan, untuk itu setiap komponen entitas (sumber atau

tujuan) ini harus diberi nama sesuai dengan lingkungan luar yang mempengaruhi sistem ini.

b. Membentuk Diagram Kontek

Diagram Kontek (*Context Diagram*) adalah diagram level tertinggi dari DAD yang menggambarkan hubungan sistem dengan lingkungan luarnya (Hollander. *et.al.* 2000). Diagram kontek ini menggambarkan satu kesatuan proses secara keseluruhan. Jika terdapat banyak arus data dalam Diagram kontek, dapat diberikan kode angka atau abjad dan kemudian diberikan penjelasan. Istilah lain dari Diagram kontek adalah DAD Level 0. Gambar 4 memperlihatkan Diagram Kontek Sistem Pesanan Makanan.



Gambar 4. Diagram Kontek Sistem Pesanan Makanan

c. Membentuk DAD Level 1 dan Level Selanjutnya

DAD level 1 ini merupakan dekomposisi dari Diagram Kontek (*Context Diagram*). DAD level 2 merupakan dekomposisi dari DAD level 1, dan seterusnya. Adapun rambu-rambu yang perlu diperhatikan menurut Celko, 1987 dalam Jeffrey *et.al.*, 2002, sebagai berikut;

Proses (*Process*)

- Jika objek berupa keluaran (*output*), itu berarti entitas sumber (*source*).
- Jika objek berupa masukan (*input*), itu berarti entitas tujuan (*sink*).
- Konsistensi dalam pemberian nama proses data.

Media Penyimpanan (*Data Storage*)

- Data dalam media penyimpanan tidak boleh berpindah langsung tanpa melalui suatu proses.
- Data tidak boleh berpindah dari entitas sumber (*source*) ke media penyimpanan tanpa melalui suatu proses.
- Data tidak boleh berpindah dari entitas tujuan (*sink*) ke media penyimpanan tanpa melalui suatu proses.
- Konsistensi dalam pemberian nama media penyimpanan.

Entitas (*Entities*)

- Data tidak boleh berpindah langsung dari entitas sumber ke entitas tujuan tanpa melalui suatu proses.
- Konsistensi dalam pemberian nama entitas sumber maupun entitas tujuan.

Arus Data (*Data Flow*)

- Arus data satu arah menggambarkan proses pembacaan data yang belum diperbaharui (*update*).
- Boleh melakukan pencabangan data dari proses yang sama.
- Setiap arus data selalu melalui proses.
- Arus data ke media penyimpanan berarti melakukan proses *update* (penghapusan atau pengeditan).
- Arus data dari media penyimpanan berarti mendapatkan data.
- Konsistensi dalam pemberian nama arus data.

DAD level 1 dan seterusnya menyediakan *view* tingkat tinggi bagi sistem dengan mendekomposisi DAD setiap level sebelumnya, seperti mengidentifikasi setiap arus informasi (keluar dan masuknya) dengan merujuk pada setiap proses dan media penyimpanan.

Gambar 5, 6, 7 dan 8 memperlihatkan DAD level 1 beserta dekomposisinya.

MENGUKUR VALIDITAS DAD

Hal yang terpenting dalam membangun DAD adalah validitas data yang terjamin. Untuk itu perlu lebih mengetahui seluk beluk permasalahan sistem secara spesifik.

Dalam hal pemanfaatan komponen-komponen dan arus informasi, (Donald S. Le Vie, Jr. 2000) memberikan skenario rambu validitas DAD yaitu;

- Adanya proses timbal balik dengan arus data disetiap proses entitas.
- Adanya proses timbal balik dengan arus data dengan media penyimpanan.
- Proses-proses tersebut dilakukan dengan simultan.

Skenario yang ada tersebut dapat kita transformasikan menjadi beberapa kaidah, yaitu;

- Semakin sedikit adanya persilangan data pada DAD, maka makin baik DAD tersebut.
- Balancing antara alur data yang masuk/keluar dari suatu level harus sama dengan alur data yang masuk/keluar pada level berikutnya. Dengan bahasa yang lain, jumlah alur data global pada suatu level harus sama pada level berikutnya.

KEUNTUNGAN DAN KELEMAHAN DAD

Keuntungan

Perancangan sistem yang beorientasi objek banyak menggunakan DAD, cepat dan relatif mudah digunakan dalam mendisain sistem serta cukup mudah sebagai jembatan penterjemah kepada pihak sekalipun bagi orang awam.

Kelemahan

DAD akan mengalami kesulitan untuk digunakan dalam perancangan sistem berskala besar.

KESIMPULAN

1. Diagram Arus Data (DAD) adalah salah satu teknik yang efektif dalam menggambarkan arus informasi dengan diagram.
2. DAD juga digunakan sebagai langkah awal yang digunakan oleh para perancang sistem untuk merepresentasikan kebutuhan sistem.
3. DAD mempunyai empat komponen utama yaitu entitas, proses, media penyimpanan dan arus data yang digunakan dalam merepresentasikan sistem.

DAFTAR PUSTAKA

Donald S. Le Vie, Jr. 2000. *Understanding Data Flow Diagrams.*

Gelinas U.J, Jr., Sutton S.G., Jane F. 2004. *Business Processes and Information Technology.* Thomson Learning.

Hollander A.S., Denna E.L., Cherrington J.O. 2000. *Accounting, Information Technology, And Business Solutions, Second Edition.* McGraw-Hill.

James Martin and Carma McClure. 1988. *Structure Techniques: The Basis for CASE.* Prentice Hall, Inc. New Jersey 07458.

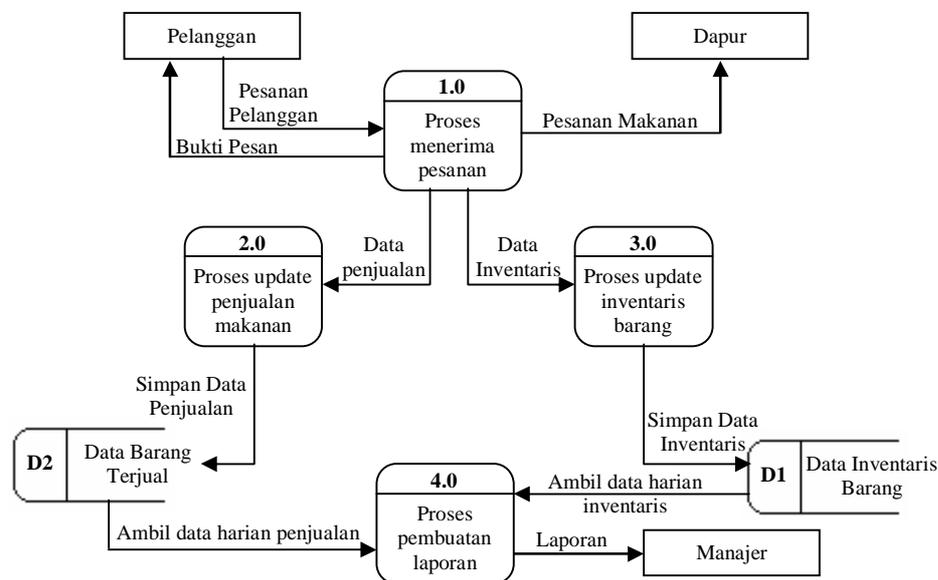
Jeffrey A.H., Joey F.G., Joseph S.V. 2002. *Modern Systems Analysis & Design third edition.* Prentice Hall, Inc. Upper Saddle River, New Jersey 07458.

Kendall & Kendall. 2002. *Systems Analysis and Design, Fifth Edition.* Prentice Hall, Inc. Upper Saddle River, New Jersey 07458

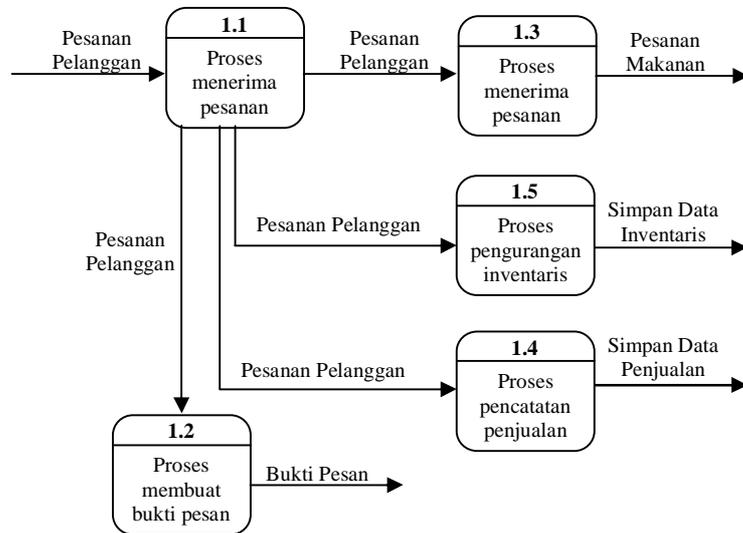
McLeod R., Jr. 1998. *Management Information Systems: Seventh Edition.* Prentice Hall, Inc. Upper Saddle River, New Jersey 07458

Ryan K. Stephens and Ronald R. Plew. 2001. *Database Design,* Sams Publishing. 46290. USA.

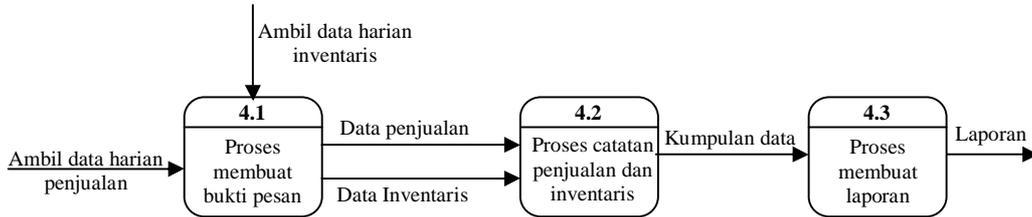
Lampiran



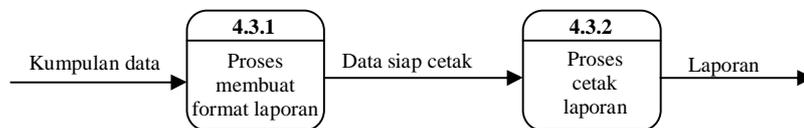
Gambar 5. DAD Level 1



Gambar 6. Dekomposisi proses 1



Gambar 7. Dekomposisi proses 4



Gambar 8. Dekomposisi proses 4.3